

Relatório - Projeto SISis

TeSP - Programação de Sistemas de Informação

Serviços e Interoperabilidade de Sistemas

Iuri Carrasqueiro, 2201127 - Gabriel Silva, 2201133

Mário Fernandes

Índice

Introdução	3
API REST	4
Endpoints	5
Ciclismo	5
User	5
Publicação	5
Comentário	5
Gosto	6
Exemplos	7
Ciclismo	7
User	8
Publicação	10
Comentário	11
Gosto	12
Messaging	14
Conclusão	15

Introdução

No âmbito da unidade curricular de Serviços e Interoperabilidade de Sistemas, este documento serve para descrever a API REST, indicar todos os *endpoints* associados e mostrar um exemplo de como invocar cada método através da linha de comandos utilizando a ferramenta cURL.

Além disso, damos uma breve explicação das funcionalidades baseadas em messaging (sistema Mosquitto) implementadas no projeto, dando a conhecer em que casos é que estas são utilizadas e como funcionam.

API REST

A API REST do projeto Ciclodias permite o acesso ao sistema do projeto, neste caso, a Framework Yii2, permitindo a uma aplicação aceder a todas as funcionalidades do mesmo.

A API tem o suporte de registar novos utilizadores, de fazer login dos mesmos e fazer a gestão destes. Além disso, tem a possibilidade de aceder a toda a informação do feed de notícias, como publicações, comentários e gostos. Todas estas funcionalidades são também acedidas por outras partes do sistema, por exemplo, o Website.

Contudo, a API apresenta o único ponto de acesso à criação de novas sessões de treino, uma vez que, estas necessitam de dados onde só um dispositivo móvel com acesso a localização GPS consegue fornecer, como por exemplo, um telemóvel *Android*. A gestão destes, ou seja, a alteração e eliminação de dados, também é feita pelo acesso da API, mas com a possibilidade de ser acedidas por outros sistemas.

Em termos de controlo de autorização, todos os métodos, menos o de login e registo, necessitam de um *token* de acesso do utilizador para permitir o acesso. Além disso, vários métodos contêm verificações, utilizando um sistema RBAC, para controlar o acesso a diferentes tipos de utilizadores.

Para esta API funcionar, foram criados 5 controladores com métodos CRUD, alguns com métodos desativados por não enquadrar com o sistema, num total de 19 métodos, mais 8 métodos Custom.

Endpoints

Ciclismo

Criar Treino - POST <http://localhost:8080/v1/ciclismo>
Ver Treinos - GET <http://localhost:8080/v1/ciclismo>
Ver Treino - GET <http://localhost:8080/v1/ciclismo/x>
Alterar Treino - PUT <http://localhost:8080/v1/ciclismo/x>
Apagar Treino - DELETE <http://localhost:8080/v1/ciclismo/x>
Sincronizar Treinos - POST <http://localhost:8080/v1/ciclismo/sync>

User

Registrar Utilizador - POST <http://localhost:8080/v1/registro/signup>
Login Utilizador - POST <http://localhost:8080/v1/login/login>
Ver Perfil - GET <http://localhost:8080/v1/user/x>
Alterar Perfil - PUT <http://localhost:8080/v1/user/x>
Apagar Perfil - DELETE <http://localhost:8080/v1/user/x>

Publicação

Criar Publicação - POST <http://localhost:8080/v1/publicacao>
Ver Publicação - GET <http://localhost:8080/v1/publicacao/x>
Ver Publicações - GET <http://localhost:8080/v1/publicacao>
Ver Publicações Utilizador - GET <http://localhost:8080/v1/publicacao/user>
Apagar Publicação - DELETE <http://localhost:8080/v1/publicacao/x>

Comentário

Criar Comentário - POST <http://localhost:8080/v1/comentario>
Ver Comentários - GET <http://localhost:8080/v1/comentario>
Ver Comentários Pub -
GET <http://localhost:8080/v1/comentario/getcomentpub/x>
Ver Comentário - GET <http://localhost:8080/v1/comentario/x>

Editar Comentário - PUT <http://localhost:8080/v1/comentario/x>

Apagar Comentário - DELETE <http://localhost:8080/v1/comentario/x>

Gosto

Criar Gosto - POST <http://localhost:8080/v1/gosto>

Ver Gostos Publicação -

GET <http://localhost:8080/v1/gosto/numgostospub/x>

Ver Gostos cada Publicação -

GET <http://localhost:8080/v1/gosto/numgostos>

Ver Gostos Publicação User -

GET <http://localhost:8080/v1/gosto/numgostosuser>

Apagar Gosto - DELETE <http://localhost:8080/v1/gosto/x>

Exemplos

Ciclismo

Criar Treino - Método CRUD responsável por adicionar uma nova sessão de treino do utilizador à base de dados;

```
curl --location --request POST 'http://localhost:8080/v1/ciclismo/access-token=1L4wPH10LNCOP5JC3dzt7xRQJ50KwK6' \
--header 'Content-Type: application/json' \
--data-raw '{
  "nome_percurso": "Percurso API",
  "duracao": 1029,
  "distancia": 4292,
  "velocidade_media": 7.45,
  "velocidade_maxima": 20.4,
  "rota": null,
  "velocidade_grafico": null
}'
{"success":true,"mensagem":null,"ciclismo":{"nome_percurso":"Percurso API","duracao":1029,"distancia":4292,"velocidade_media":7.45,"velocidade_maxima":20.4,"velocidade_grafico":null,"rota":null,"data_treino":"2022-01-08 11:23:13","user_id":5,"id":6}}
```

Ver Treinos - Método CRUD que devolve todos os treinos realizados pelo utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/ciclismo/access-token=1L4wPH10LNCOP5JC3dzt7xRQJ50KwK6' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"ciclismo":[{"id":4,"nome_percurso":"Percurso Torres Vedras","duracao":6012,"distancia":900,"velocidade_media":10.1,"velocidade_maxima":20.4,"velocidade_grafico":null,"rota":null,"data_treino":"2021-12-21 14:30:01","user_id":5}, {"id":5,"nome_percurso":"Percurso com rota e grafico","duracao":71,"distancia":428,"velocidade_media":21.8457,"velocidade_maxima":48.0372,"velocidade_grafico":[{"x":224417.19,"y":77765.15,"z":23029.11,"v":56545.8,"a":203097.3,"b":8844825.0,"c":0.0,"d":0.2,"t":7374112.7,"e":42249.14,"f":15195.19,"g":520959.21,"h":268980.19,"i":238097.19,"j":802483.22,"k":474035.26,"l":775343.29,"m":813385.31,"n":493929.32,"o":80445.32,"p":99885.33,"q":265575.34,"r":875.31,"s":36.368434,"t":36.559517,"u":38.62568,"v":213074.39,"w":9193.48,"x":037235.39,"y":63434.37,"z":6049.36,"aa":59412.34,"ab":742126.32,"ac":638348.30,"ad":24443.29,"ae":117117.29,"af":866805.28,"ag":794852.27,"ah":77208.26,"ai":385626.24,"aj":503798.22,"ak":173983.21,"al":322756.17,"am":119686.14,"an":7273.96,"ao":36.86434,"ap":14.782849,"aq":16.80632,"ar":19.02666,"as":22.143686,"at":24.053917,"au":24.779722,"av":28.985506,"aw":19.446732,"ax":16.876976,"ay":14.63836,"az":12.962649,"ba":12.847327,"bb":12.199803,"bc":8.113976,"bd":3.779025,"be":4.2293496,"bf":7.09468,"bg":11.209326,"bh":17.265697,"bi":21.970772,"bj":24.13636,"bk":846165.32,"bl":202667.37,"bm":9653177,"bn":null,"bo":null,"bp":null,"bq":null,"br":null,"bs":null,"bt":null,"bu":null,"bv":null,"bw":null,"bx":null,"by":null,"bz":null,"ca":null,"cb":null,"cc":null,"cd":null,"ce":null,"cf":null,"cg":null,"ch":null,"ci":null,"cj":null,"ck":null,"cl":null,"cm":null,"cn":null,"co":null,"cp":null,"cq":null,"cr":null,"cs":null,"ct":null,"cu":null,"cv":null,"cw":null,"cx":null,"cy":null,"cz":null,"da":null,"db":null,"dc":null,"dd":null,"de":null,"df":null,"dg":null,"dh":null,"di":null,"dj":null,"dk":null,"dl":null,"dm":null,"dn":null,"do":null,"dp":null,"dq":null,"dr":null,"ds":null,"dt":null,"du":null,"dv":null,"dw":null,"dx":null,"dy":null,"dz":null,"ea":null,"eb":null,"ec":null,"ed":null,"ee":null,"ef":null,"eg":null,"eh":null,"ei":null,"ej":null,"ek":null,"el":null,"em":null,"en":null,"eo":null,"ep":null,"eq":null,"er":null,"es":null,"et":null,"eu":null,"ev":null,"ew":null,"ex":null,"ey":null,"ez":null,"fa":null,"fb":null,"fc":null,"fd":null,"fe":null,"ff":null,"fg":null,"fh":null,"fi":null,"fj":null,"fk":null,"fl":null,"fm":null,"fn":null,"fo":null,"fp":null,"fq":null,"fr":null,"fs":null,"ft":null,"fu":null,"fv":null,"fw":null,"fx":null,"fy":null,"fz":null,"ga":null,"gb":null,"gc":null,"gd":null,"ge":null,"gf":null,"gg":null,"gh":null,"gi":null,"gj":null,"gk":null,"gl":null,"gm":null,"gn":null,"go":null,"gp":null,"gq":null,"gr":null,"gs":null,"gt":null,"gu":null,"gv":null,"gw":null,"gx":null,"gy":null,"gz":null,"ha":null,"hb":null,"hc":null,"hd":null,"he":null,"hf":null,"hg":null,"hh":null,"hi":null,"hj":null,"hk":null,"hl":null,"hm":null,"hn":null,"ho":null,"hp":null,"hq":null,"hr":null,"hs":null,"ht":null,"hu":null,"hv":null,"hw":null,"hx":null,"hy":null,"hz":null,"ia":null,"ib":null,"ic":null,"id":null,"ie":null,"if":null,"ig":null,"ih":null,"ii":null,"ij":null,"ik":null,"il":null,"im":null,"in":null,"io":null,"ip":null,"iq":null,"ir":null,"is":null,"it":null,"iu":null,"iv":null,"iw":null,"ix":null,"iy":null,"iz":null,"ja":null,"jb":null,"jc":null,"jd":null,"je":null,"jf":null,"jg":null,"jh":null,"ji":null,"jj":null,"jk":null,"jl":null,"jm":null,"jn":null,"jo":null,"jp":null,"jq":null,"jr":null,"js":null,"jt":null,"ju":null,"jv":null,"jw":null,"jx":null,"jy":null,"jz":null,"ka":null,"kb":null,"kc":null,"kd":null,"ke":null,"kf":null,"kg":null,"kh":null,"ki":null,"kj":null,"kk":null,"kl":null,"km":null,"kn":null,"ko":null,"kp":null,"kq":null,"kr":null,"ks":null,"kt":null,"ku":null,"kv":null,"kw":null,"kx":null,"ky":null,"kz":null,"la":null,"lb":null,"lc":null,"ld":null,"le":null,"lf":null,"lg":null,"lh":null,"li":null,"lj":null,"lk":null,"ll":null,"lm":null,"ln":null,"lo":null,"lp":null,"lq":null,"lr":null,"ls":null,"lt":null,"lu":null,"lv":null,"lw":null,"lx":null,"ly":null,"lz":null,"ma":null,"mb":null,"mc":null,"md":null,"me":null,"mf":null,"mg":null,"mh":null,"mi":null,"mj":null,"mk":null,"ml":null,"mm":null,"mn":null,"mo":null,"mp":null,"mq":null,"mr":null,"ms":null,"mt":null,"mu":null,"mv":null,"mw":null,"mx":null,"my":null,"mz":null,"na":null,"nb":null,"nc":null,"nd":null,"ne":null,"nf":null,"ng":null,"nh":null,"ni":null,"nj":null,"nk":null,"nl":null,"nm":null,"nn":null,"no":null,"np":null,"nq":null,"nr":null,"ns":null,"nt":null,"nu":null,"nv":null,"nw":null,"nx":null,"ny":null,"nz":null,"oa":null,"ob":null,"oc":null,"od":null,"oe":null,"of":null,"og":null,"oh":null,"oi":null,"oj":null,"ok":null,"ol":null,"om":null,"on":null,"oo":null,"op":null,"oq":null,"or":null,"os":null,"ot":null,"ou":null,"ov":null,"ow":null,"ox":null,"oy":null,"oz":null,"pa":null,"pb":null,"pc":null,"pd":null,"pe":null,"pf":null,"pg":null,"ph":null,"pi":null,"pj":null,"pk":null,"pl":null,"pm":null,"pn":null,"po":null,"pp":null,"pq":null,"pr":null,"ps":null,"pt":null,"pu":null,"pv":null,"pw":null,"px":null,"py":null,"pz":null,"qa":null,"qb":null,"qc":null,"qd":null,"qe":null,"qf":null,"qg":null,"qh":null,"qi":null,"qj":null,"qk":null,"ql":null,"qm":null,"qn":null,"qo":null,"qp":null,"qq":null,"qr":null,"qs":null,"qt":null,"qu":null,"qv":null,"qw":null,"qx":null,"qy":null,"qz":null,"ra":null,"rb":null,"rc":null,"rd":null,"re":null,"rf":null,"rg":null,"rh":null,"ri":null,"rj":null,"rk":null,"rl":null,"rm":null,"rn":null,"ro":null,"rp":null,"rq":null,"rr":null,"rs":null,"rt":null,"ru":null,"rv":null,"rw":null,"rx":null,"ry":null,"rz":null,"sa":null,"sb":null,"sc":null,"sd":null,"se":null,"sf":null,"sg":null,"sh":null,"si":null,"sj":null,"sk":null,"sl":null,"sm":null,"sn":null,"so":null,"sp":null,"sq":null,"sr":null,"ss":null,"st":null,"su":null,"sv":null,"sw":null,"sx":null,"sy":null,"sz":null,"ta":null,"tb":null,"tc":null,"td":null,"te":null,"tf":null,"tg":null,"th":null,"ti":null,"tj":null,"tk":null,"tl":null,"tm":null,"tn":null,"to":null,"tp":null,"tq":null,"tr":null,"ts":null,"tt":null,"tu":null,"tv":null,"tw":null,"tx":null,"ty":null,"tz":null,"ua":null,"ub":null,"uc":null,"ud":null,"ue":null,"uf":null,"ug":null,"uh":null,"ui":null,"uj":null,"uk":null,"ul":null,"um":null,"un":null,"uo":null,"up":null,"uq":null,"ur":null,"us":null,"ut":null,"uu":null,"uv":null,"uw":null,"ux":null,"uy":null,"uz":null,"va":null,"vb":null,"vc":null,"vd":null,"ve":null,"vf":null,"vg":null,"vh":null,"vi":null,"vj":null,"vk":null,"vl":null,"vm":null,"vn":null,"vo":null,"vp":null,"vq":null,"vr":null,"vs":null,"vt":null,"vu":null,"vv":null,"vw":null,"vx":null,"vy":null,"vz":null,"wa":null,"wb":null,"wc":null,"wd":null,"we":null,"wf":null,"wg":null,"wh":null,"wi":null,"wj":null,"wk":null,"wl":null,"wm":null,"wn":null,"wo":null,"wp":null,"wq":null,"wr":null,"ws":null,"wt":null,"wu":null,"wv":null,"ww":null,"wx":null,"wy":null,"wz":null,"xa":null,"xb":null,"xc":null,"xd":null,"xe":null,"xf":null,"xg":null,"xh":null,"xi":null,"xj":null,"xk":null,"xl":null,"xm":null,"xn":null,"xo":null,"xp":null,"xq":null,"xr":null,"xs":null,"xt":null,"xu":null,"xv":null,"xw":null,"xx":null,"xy":null,"xz":null,"ya":null,"yb":null,"yc":null,"yd":null,"ye":null,"yf":null,"yg":null,"yh":null,"yi":null,"yj":null,"yk":null,"yl":null,"ym":null,"yn":null,"yo":null,"yp":null,"yq":null,"yr":null,"ys":null,"yt":null,"yu":null,"yv":null,"yw":null,"yx":null,"yy":null,"yz":null,"za":null,"zb":null,"zc":null,"zd":null,"ze":null,"zf":null,"zg":null,"zh":null,"zi":null,"zj":null,"zk":null,"zl":null,"zm":null,"zn":null,"zo":null,"zp":null,"zq":null,"zr":null,"zs":null,"zt":null,"zu":null,"zv":null,"zw":null,"zx":null,"zy":null,"zz":null}]]
```

Ver Treino - Método CRUD que devolve um treino a pedido do utilizador, com verificação de se o treino pertence ao utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/ciclismo/67access-token=1L4wPH10LNCOP5JC3dzt7xRQJ50KwK6' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"ciclismo":{"id":6,"nome_percurso":"Percurso API","duracao":1029,"distancia":4292,"velocidade_media":7.45,"velocidade_maxima":20.4,"velocidade_grafico":null,"rota":null,"data_treino":"2022-01-08 11:23:13","user_id":5}}
```

Alterar Treino - Método CRUD que altera uma sessão de treino, a pedido do utilizador. Este método também contém verificação de se o treino pertence ao utilizador;

```
curl --location --request PUT 'http://localhost:8080/v1/ciclismo/67access-token=1L4wPH10LNCOP5JC3dzt7xRQJ50KwK6' \
--header 'Content-Type: application/json' \
--data-raw '{
  "nome_percurso": "Percurso API alterado"
}'
{"success":true,"mensagem":null,"ciclismo":{"id":6,"nome_percurso":"Percurso API alterado","duracao":1029,"distancia":4292,"velocidade_media":7.45,"velocidade_maxima":20.4,"velocidade_grafico":null,"rota":null,"data_treino":"2022-01-09 15:35:26","user_id":5}}
```

Apagar Treino - Método CRUD que elimina um treino da base de dados, a pedido do utilizador. Também este contém verificação;

```
curl --location --request DELETE 'http://localhost:8080/v1/ciclisismo/6?access-token=IL4wPHi0LMCGP5Jc3dzT7xRqJSOKbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"ciclisismo":null}
```

Sincronizar Treinos - Método Custom que recebe um array de sessões de treino, adiciona-as na base de dados e devolve todos os treinos realizados pelo utilizador;

[illegible]

User

Registrar Utilizador - Método Custom que regista um novo utilizador na base de dados;

```
curl --location --request POST 'http://localhost:8080/v1/registo/signup' \
--header 'Content-Type: application/json' \
--data-raw '{
  "username": "luis",
  "email": "luis@gmail.com",
  "password": "123456789",
  "primeiro_nome": "Luis",
  "ultimo_nome": "Torres"
}'
{"success":true}%
```


Login Utilizador - Método Custom que faz o login de um utilizador, devolvendo o token de acesso para futuros pedidos API e informação sobre o perfil;

```
curl --location --request POST 'http://localhost:8080/v1/login/login' \
--header 'Content-Type: application/json' \
--data-raw '{
  "username": "luis",
  "password": "123456789"
}'
{"success":true,"token":"vUArG7SLrq9ZjCxoHfhrqIMBsH00vjNR","id":6,"primeiro_nome":"Luis","ultimo_nome":"Torres","mensagem":null}
```

Ver Perfil - Método CRUD que devolve informações sobre o perfil do um utilizador, a pedido do mesmo. Este método contém verificação de se o perfil pertence ao utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/user/6?access-token=lu0SHi1X3Gcpgjv2tGN0yv8huNRLtt6' \
--header 'Content-Type: application/json' \
--data-raw ''
{"success":true,"mensagem":null,"primeiro_nome":"Luis","ultimo_nome":"Torres","data_nascimento":"nulo"}
```

Alterar Perfil - Método CRUD que altera as informações de perfil do um utilizador, a pedido do mesmo. Este método também contém verificação de se o perfil pertence ao utilizador;

```
curl --location --request PUT 'http://localhost:8080/v1/user/6?access-token=_Di0cqyp-Mmnj0DBDuGFXFyA98xUc5-M' \
--header 'Content-Type: application/json' \
--data-raw '{
  "primeiro_nome": "Torres",
  "ultimo_nome": "Luis",
  "data_nascimento": "1991-09-20"
}'
{"success":true,"mensagem":null,"primeiro_nome":"Torres","ultimo_nome":"Luis","data_nascimento":"1991-09-20"}
```

Apagar Perfil - Método CRUD que apaga um utilizador, juntamente com todos os seus treinos e publicações, a pedido do utilizador. Este método também contém verificação;

```
curl --location --request DELETE 'http://localhost:8080/v1/user/6?access-token=lu0SHi1X3Gcpgjv2tGN0yv8huNRLtt6' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"primeiro_nome":null,"ultimo_nome":null,"data_nascimento":null}
```

Publicação

Criar Publicação - Método CRUD que adiciona uma nova publicação à base de dados, a partir de um ciclismo. Este método contém verificação de se o treino que irá ser publicado é do utilizador;

```
curl --location --request POST 'http://localhost:8080/v1/publicacao?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjSOKbKG' \
--header 'Content-Type: application/json' \
--data-raw '{
  "ciclismo_id": 5
}'
{"success":true,"mensagem":null,"publicacao":{"ciclismo_id":5,"createtime":"2022-01-09 12:44:04","id":4}}
```

Ver Publicação - Método CRUD que devolve uma publicação, a pedido do utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/publicacao/3?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjSOKbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"publicacao":{"id":3,"createtime":"2022-01-08 11:28:52","ciclismo_id":5}}
```

Ver Publicações - Método CRUD que devolve todas as publicações;

```
curl --location --request GET 'http://localhost:8080/v1/publicacao?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjSOKbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"publicacao":[{"id":3,"createtime":"2022-01-08 11:28:52","ciclismo_id":5}, {"id":2,"createtime":"2021-12-25 01:27:09","ciclismo_id":3}, {"id":1,"createtime":"2021-12-22 19:10:30","ciclismo_id":1}]}
```

Ver Publicações Utilizador - Método Custom que devolve todas as publicações feitas pelo utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/publicacao/user?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjSOKbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"publicacao":[{"id":3,"createtime":"2022-01-08 11:28:52","ciclismo_id":5}]}
```

Apagar Publicação - Método CRUD que elimina uma publicação, a pedido do utilizador. Este método inclui verificação de se a publicação pertence ao utilizador ou se o utilizador é moderador;

```
curl --location --request DELETE 'http://localhost:8080/v1/publicacao/3?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjSOKbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"publicacao":null}
```

Comentário

Criar Comentário - Método CRUD responsável por adicionar um novo comentário do utilizador à base de dados, a partir de uma publicação;

```
curl --location --request POST 'http://localhost:8080/v1/comentario?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj50KbKG' \
--header 'Content-Type: application/json' \
--data-raw '{
  "publicacao_id": 2,
  "content": "Comentei"
}'
{"success":true,"mensagem":null,"comentario":{"publicacao_id":2,"user_id":5,"createtime":"2022-01-08 11:33:41","content":"Comentei","id":6}}
```

Ver Comentários - Método CRUD que devolve todos os comentários que existem na base de dados;

```
curl --location --request GET 'http://localhost:8080/v1/comentario/5?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj50KbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"comentario":{"id":5,"content":"Comentei","createtime":"2022-01-08 11:31:52","publicacao_id":2,"user_id":5}}
```

Ver Comentários Pub - Método Custom que devolve todos os comentários de uma publicação, a pedido do utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/comentario/getcomentpub/2?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj50KbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"comentario":[{"id":4,"content":"Bom Treino!",,"createtime":"2021-12-26 23:11:51","publicacao_id":2,"user_id":3},{id":5,"content":"Comentei","createtime":"2022-01-08 11:31:52","publicacao_id":2,"user_id":5},{id":6,"content":"Comentei","createtime":"2022-01-08 11:33:41","publicacao_id":2,"user_id":5}]}
```

Ver Comentário - Método CRUD que devolve um comentário;

```
curl --location --request GET 'http://localhost:8080/v1/comentario/5?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj50KbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"comentario":{"id":5,"content":"Comentei","createtime":"2022-01-08 11:31:52","publicacao_id":2,"user_id":5}}
```

Editar Comentário - Método CRUD que altera um comentário, a pedido do utilizador. Este método contém verificação de se o comentário pertence ao utilizador;

```
curl --location --request PUT 'http://localhost:8080/v1/comentario/5?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj50KbKG' \
--header 'Content-Type: application/json' \
--data-raw '{
  "content": "Comentei outra vez"
}'
{"success":true,"mensagem":null,"comentario":{"id":5,"content":"Comentei outra vez (Editado)","createtime":"2022-01-08 11:31:52","publicacao_id":2,"user_id":5}}
```

Apagar Comentário - Método CRUD que elimina um comentário da base de dados. Este método verifica se o utilizador ter permissão de apagar o comentário (se este for o criador deste) e/ou se ele é moderador;

```
curl --location --request DELETE 'http://localhost:8080/v1/comentario/5?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj5OKbKG' \
--header 'Content-Type: application/json' \
{"success":true,"mensagem":null,"comentario":null}%
```

Gosto

Criar Gosto - Método CRUD que adiciona um novo gosto do utilizador à base de dados, a partir de uma publicação;

```
curl --location --request POST 'http://localhost:8080/v1/gosto?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj5OKbKG' \
--header 'Content-Type: application/json' \
--data-raw '{
  "publicacao_id": 2
}'
{"success":true,"mensagem":null,"gosto":{"user_id":5,"publicacao_id":2,"id":5}}%
```

Ver Gostos Publicação - Método Custom que devolve o número de gostos de uma publicação, a pedido do utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/gosto/numgostopub/1?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj5OKbKG' \
--header 'Content-Type: application/json' \
{"success":true,"mensagem":null,"gosto":{"publicacao_id":1,"count":3}}%
```

Ver Gostos cada Publicação - Método Custom que devolver o número de gostos de todas as publicações que existem na base de dados;

```
curl --location --request GET 'http://localhost:8080/v1/gosto/numgostos?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQj5OKbKG' \
--header 'Content-Type: application/json' \
{"success":true,"mensagem":null,"gosto":[{"publicacao_id":1,"count":3}, {"publicacao_id":2,"count":2}]}%
```

Ver Gostos Publicação User - Método Custom que devolve o número de gostos de todas as publicações do utilizador;

```
curl --location --request GET 'http://localhost:8080/v1/gosto/numgostosuser?access-token=wpvSoVPduEh99HYs95yZo_0P6tETHEPB' \
--header 'Content-Type: application/json' \
{"success":true,"mensagem":null,"gosto":[{"publicacao_id":2,"count":2}]}%
```

Apagar Gosto - Método CRUD que elimina um gosto da base de dados, a pedido do utilizador. Este método verifica se o gosto foi adicionado pelo utilizador que, só assim, é que o método elimina o gosto;

```
curl --location --request DELETE 'http://localhost:8080/v1/gosto/5?access-token=IL4wPHi0LMCGPs5Jc3dzt7xRQjS0KbKG' \
--header 'Content-Type: application/json'
{"success":true,"mensagem":null,"gosto":null}
```

Messaging

Em termos de funcionalidades baseadas em *messaging*, o projeto Ciclodias apresenta um sistema de notificações para os utilizadores registados na base de dados. Sempre que um novo utilizador se regista, este fica possibilitado de receber um aviso (notificação), caso uma aplicação cliente implemente este sistema de notificações.

Os casos de receber um aviso são os seguintes:

- Quando um utilizador reage a uma publicação feita por um utilizador, este recebe um aviso a indicar que um utilizador reagiu a uma das suas publicações. Este aviso é enviado no momento da criação do gosto (quando este é adicionado à base de dados);
- Quando um utilizador adiciona um comentário a uma publicação feita por um utilizador, este recebe um aviso a indicar que um utilizador comentou uma das suas publicações. Exatamente como o caso anterior, este aviso também é enviado no momento da criação do comentário;
- Caso um novo treino seja adicionado à base de dados e a duração, distância ou velocidade média deste seja o valor mais alto que irá existir na base de dados, é enviado um aviso para todos os utilizadores, a indicar que existe um novo valor mais alto em cada um destes campos.

Este aviso é importante, já que, o Website do projeto implementa um TOP 10 destes valores, com isso, este sistema avisa sempre que existir um novo utilizador no primeiro lugar.

Conclusão

Concluindo, esta vertente do projeto permitiu-nos adquirir novos conhecimentos na criação e desenvolvimento de APIs, como é que estas funcionam e as ligações destas com outras aplicações.

Além disso, também ganhamos competências no desenvolvimento de funcionalidades baseadas em *messaging*.